Chapter 9
Describing Process Specifications and Structured Decisions

Systems Analysis and Design

Kendall & Kendall

Sixth Edition

Major Topics

- Process specifications
- Business rules
- Structured English
- Decision tables
- Decision trees
- Horizontal balancing

Process Specifications

- Process specifications are created for primitive processes and some higher level processes on a data flow diagram.
- They are also called minispecs.

Goal of Creating Process Specifications

The goals of producing process specifications are:

- Reduce process ambiguity.
- Obtain a precise description of what is accomplished.
- Validate the system design, including data flow diagrams and the data dictionary.

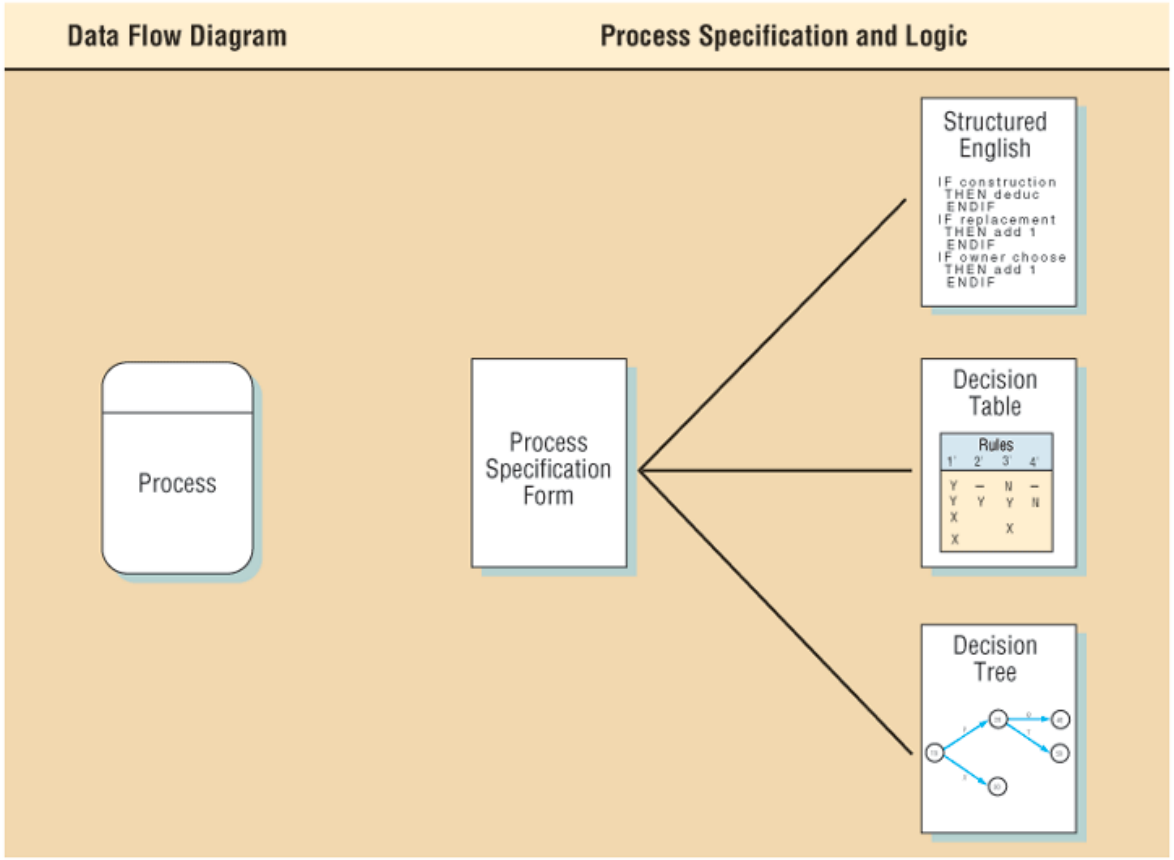
Process Specifications

Process specifications are not created for:

- Physical input and/or output processes.
- Processes that represent simple data validation.
- Processes for which prewritten code already exists.

Data Flow Diagram and Process Specifications

Figure 9.1 How process specifications relate to the data flow diagram.



Process Specification Format

- Process specifications link the process to the DFD and the data dictionary.
- The following information should be entered:
  - The process number, which must match the process ID on the data flow diagram.
  - This allows an analyst to work or review any process and easily locate the data flow diagram containing the process.

Process Specification Format (Continued)

- The process name, the same as displays within the process symbol on the DFD.
- A brief description of what the process accomplishes.
- A list of input and output data flow, using the names found on the data flow diagram.
- Data names used in the formulae or logic should match the data dictionary, for consistency and good communication.

Process Specification Format (Continued)

- An indication of the type of process, whether it is batch, online, or manual.
- All online processes require screen designs.
- All manual processes should have well-defined procedures for employees performing the process tasks.
- If the process has prewritten code for it, include the name of the subprogram or function.

Process Specification Format (Continued)

- A description of the process logic.
- This should state policy and business rules, not computer language pseudocode.
- A reference to further information, such as a structured English description, a decision table, or tree depicting the logic.
- List any unresolved issues.
- These issues form the basis of the questions used for a follow-up interview.

Business Rules

Business rules include the following:

- Definitions of business terms
- Business conditions and actions
- Data integrity constraints
- Mathematical and functional derivations
- Logical inferences
- Processing sequences
- Relationships among facts about the business

Process Specification Example Part 1

Number 1
Name Add Customer Order

Description Key and add the Customer Order.
The order should be edited for correct information.
Customer and Item master files are updated.

Input Data Flow
- Customer Order Form from the Customer
- Customer Record from data store D1, Customer Master File
- Item Record from data store D2, Item Master File

Process Specification Example Part 2

Output Data Flow

- Pending Order to data store D3, Order File
- Backordered Item Record to the Inventory Control Department
- Updated Customer and Item records

Type of process Online

Structured English

- Structured English is based on structured logic and Simple English statements such as add, multiply, move, and so on.
- It is an appropriate technique for analyzing the system when structured decisions are not complex.

Steps to Use Structured English

- The following steps are needed:
  - Express all logic in terms of sequential structures, decision structures, case structures, or iterations.
  - Use and capitalize accepted keywords such as IF, THEN, ELSE, DO, and PERFORM.
  - Indent blocks of statements to show their hierarchy (nesting) clearly.

Steps to Use Structured English (Continued)

- Underline words or phrases used have been defined in a data dictionary to signify that they have a specialized, reserved meaning.
- Be careful when using "and" and "or ”.
- Avoid confusion when using logical comparisons such as "greater than" and "greater than or equal to”.

Structured English

Figure 9.5 Examples of logic expressed in a sequential structure, a decision structure, a case structure, and an iteration.

Structured English Type	Example
Sequential Structure A block of instructions in which no branching occurs	Action #1 Action #2 Action #3
Decision Structure Only IF a condition is true, complete the following statements; otherwise, jump to the ELSE	IF Condition A is True THEN implement Action A ELSE implement Action B ENDIF
Case Structure A special type of decision structure in which the cases are mutually exclusive (if one occurs, the others cannot)	IF Case #1 implement Action #1 ELSE IF Case #2 Implement Action #2 ELSE IF Case #3 Implement Action #3 ELSE IF Case #4 Implement Action #4 ELSE print error ENDIF
Iteration Blocks of statements that are repeated until done	DO WHILE there are customers. Action #1 ENDDO

Advantages of Structured English

- Clarifying the logic and relationships found in human languages
- An effective communication tool, and easy to teach and understand

Data Dictionary and Structured English

The data dictionary is a starting point for creating structured English:

- Sequenced data dictionary entries become simple structured English statements.
- Selection [] entries become IF..THEN...ELSE statements.
- Iteration { } entries become DO WHILE, DO UNTIL, or PERFORM UNTIL structured English statements.

Decision Tables

- Decision tables provide a way to examine, describe, and document decisions using a table.
- They are used to:
  - Describe the conditions.
  - Identify possible decision alternatives.
  - Indicate actions should be performed.
  - Describe actions.

Decision Table Format

Figure 9.9 The standard format used for presenting a decision table.

Conditions and Actions	Rules
Conditions	Condition Alternatives
Actions	Action Entries

Decision Table Example

Figure 9.11 Constructing a decision table for deciding which catalog to send to customers who order only from selected catalogs.

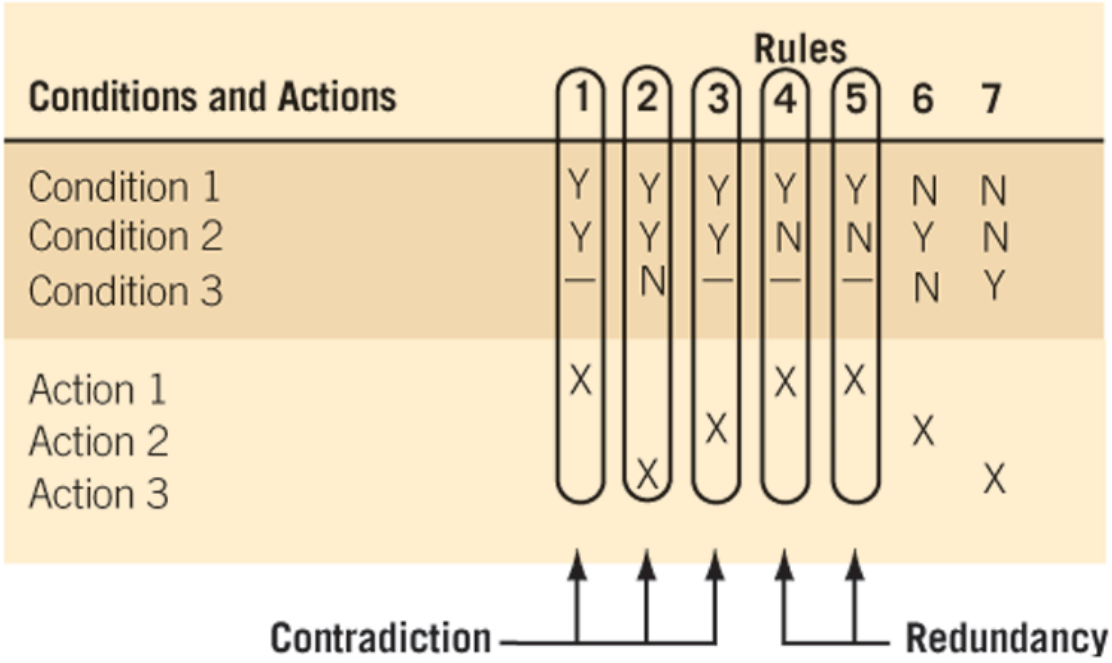
Conditions and Actions	Rules 1	2	3	4	5	6	7	8
Customer ordered from Fall catalog	Y	Y	Y	Y	N	N	N	N
Customer ordered from Christmas catalog	Y	Y	N	N	Y	Y	N	N
Customer ordered from specialty catalog	Y	N	Y	N	Y	N	Y	N
Send out this year's Christmas catalog		X		X		X		X
Send out specialty catalog			X				X	
Send out both catalogs	X				X			

Decision Tables (Continued)

- Decision tables help analysts ensure completeness and accuracy.
- Four main problems that can occur in developing decision tables:
  - Incompleteness.
  - Impossible situations.
  - Contradictions.
  - Redundancy.

Redundancy and Contradictions

Figure 9.15
Checking the decision table for inadvertent contradictions and redundancy is important.



Decision Trees

- Decision trees are used when complex branching occurs in a structured decision process.
- Trees are also useful when it is essential to keep a string of decisions in a particular sequence.

Drawing Decision Trees

- First, identify all conditions and actions and the order and timing of these (if they are critical).
- Second, begin building the tree from left to right while making sure you are complete in listing all possible alternatives before moving over to the right.

Decision Tree Example

Figure 9.18 Drawing a decision tree to show the noncash purchase approval actions for a department store.

Decision Tree Advantages

Three advantages over a decision table are:

- The order of checking conditions and executing actions is immediately noticeable.
- Second, conditions and actions of decision trees are found on some branches but not on others.
- Third, compared to decision tables, decision trees are more readily understood by others in the organization.

Selecting a Structured Decision Analysis Technique

Guidelines are as follows:

- Use structured English when there are many repetitious actions or when communication to end users is important.
- Use decision tables when complex combination of conditions, actions, and rules are found or you require a method that effectively avoids impossible situations, redundancies, and contradictions.

Selecting a Structured Decision Analysis Technique

Guidelines are as follows (continued):

- Use decision trees when the sequence of conditions and actions is critical or when not every condition is relevant to every action (the branches are different).

Parent Process Specifications

- If a process explodes to a child diagram, the process becomes a control module when the computer program representing the process is written.
- The logic of the process shows the sequence that the child diagram processes must be executed in.

Program Process Specification

- All the process specifications are consolidated for a computer program and are included in the specification packet given to the computer programmer.
- Since they are developed for one process, the logic is easier to understand.

Horizontal Balancing

- Horizontal balancing means that all output data flow must be either on input data flow or described in the process logic.
- It is used to verify that each process has the required data dictionary entries defined and the formulas and logic necessary to produce the output.

Rules for Horizontal Balancing

Rules for horizontal balancing are:

- All base elements on an output data flow must be present on an input data flow.
- All derived elements on an output data flow must be either:
  - Present on an input data flow, or
  - Created by the process.